

Correlation

- Describes direction and strength of relationship between two variables
- Set `numeric_only=True` to prevent errors with *non-numeric* columns

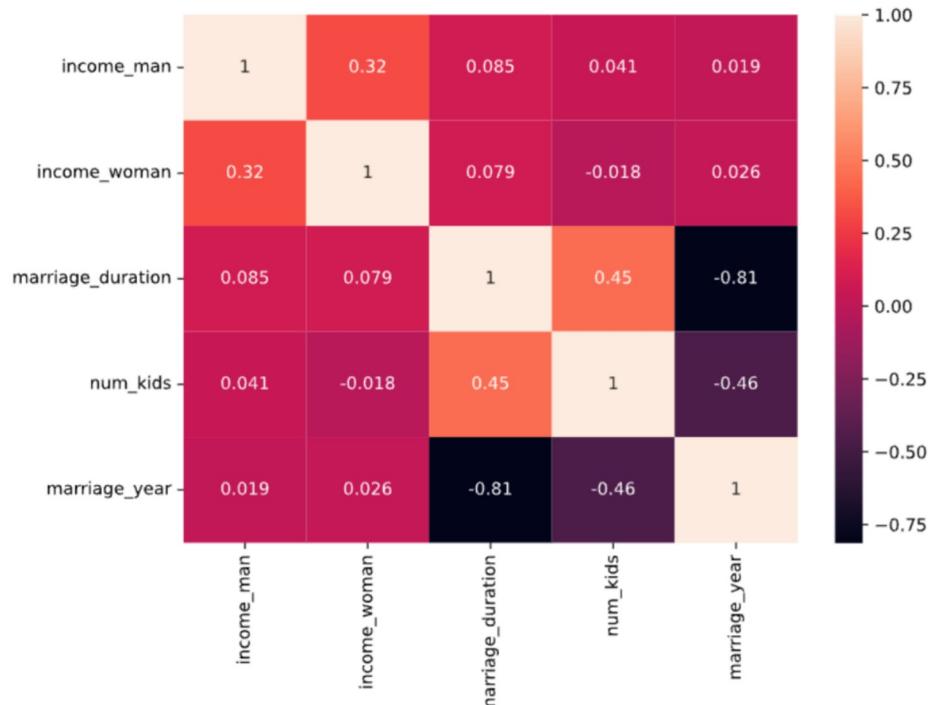
```
divorce.corr(numeric_only=True)
```

	income_man	income_woman	marriage_duration	num_kids	marriage_year
income_man	1.000	0.318	0.085	0.041	0.019
income_woman	0.318	1.000	0.079	-0.018	0.026
marriage_duration	0.085	0.079	1.000	0.447	-0.812
num_kids	0.041	-0.018	0.447	1.000	-0.461
marriage_year	0.019	0.026	-0.812	-0.461	1.000

Calculates Pearson correlation coefficient

Correlation heatmaps

```
sns.heatmap(divorce.corr(numeric_only=True), annot=True)  
plt.show()
```



Correlation in context

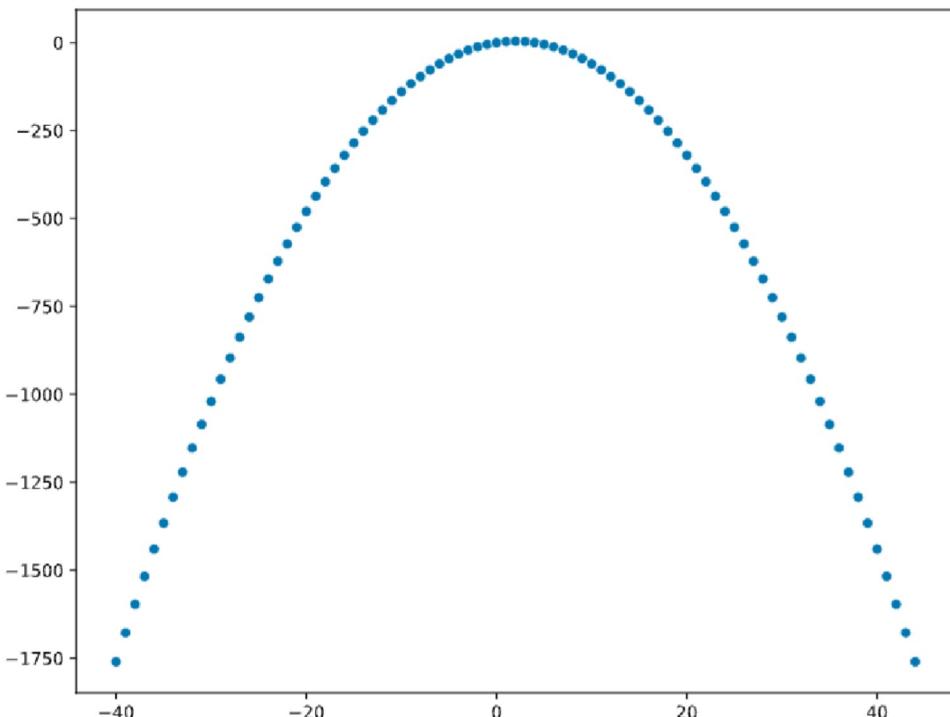
```
divorce["divorce_date"].min()
```

```
Timestamp('2000-01-08 00:00:00')
```

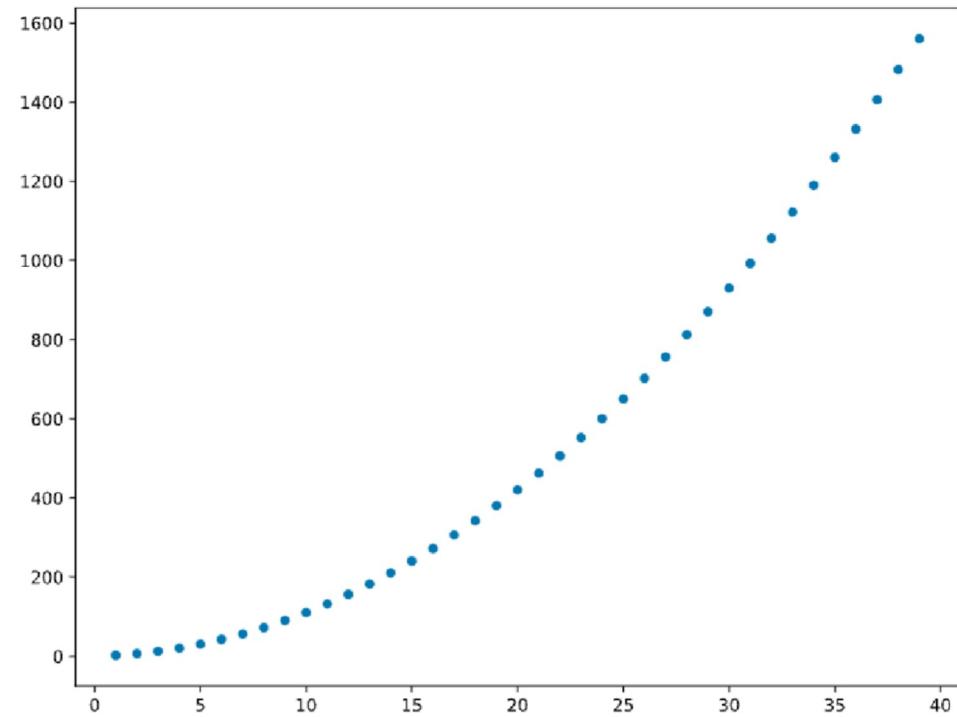
```
divorce["divorce_date"].max()
```

```
Timestamp('2015-11-03 00:00:00')
```

Visualizing relationships



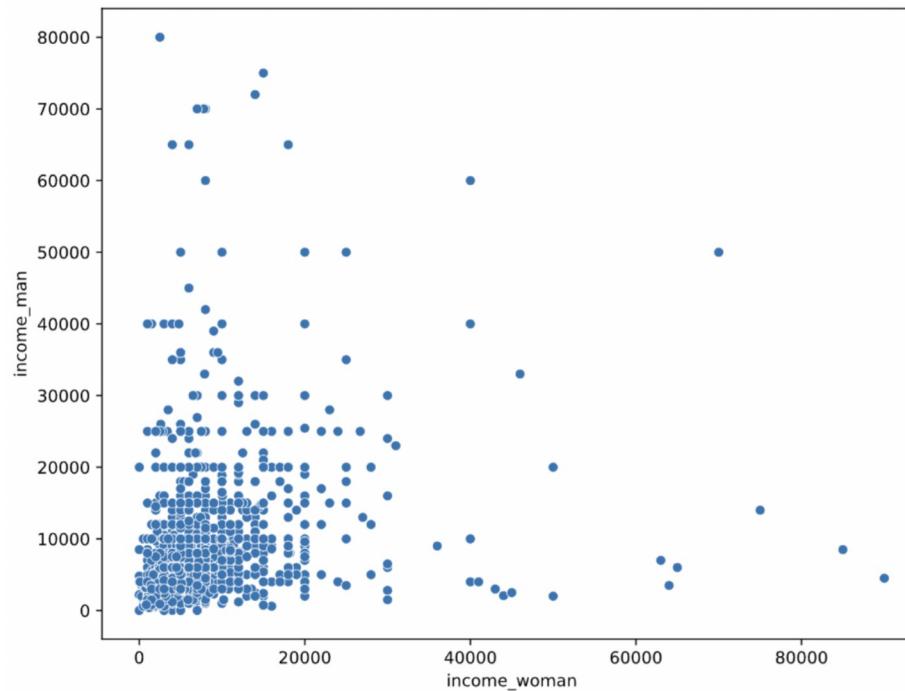
- Strong relationship—but not linear
- Pearson correlation coefficient: `-6.48e-18`



- Quadratic relationship; not linear
- Pearson correlation coefficient: `.971211`

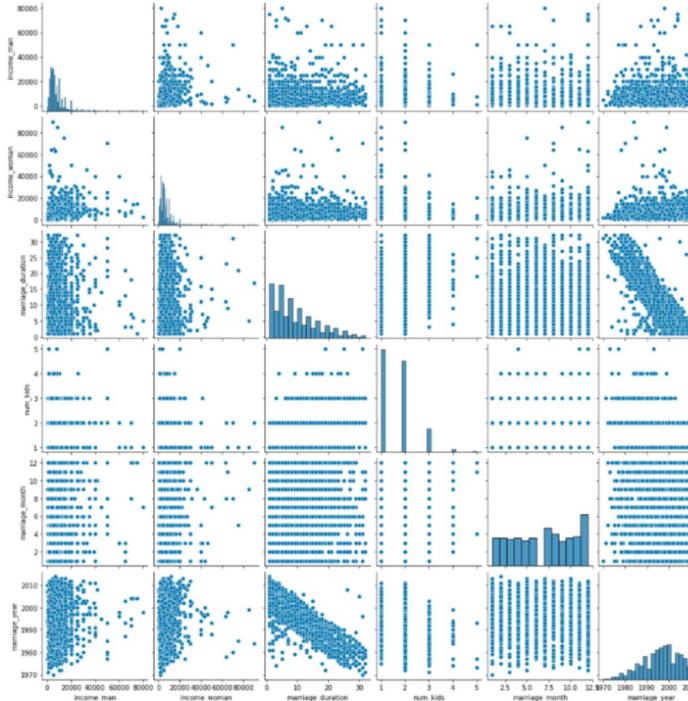
Scatter plots

```
sns.scatterplot(data=divorce, x="income_man", y="income_woman")  
plt.show()
```



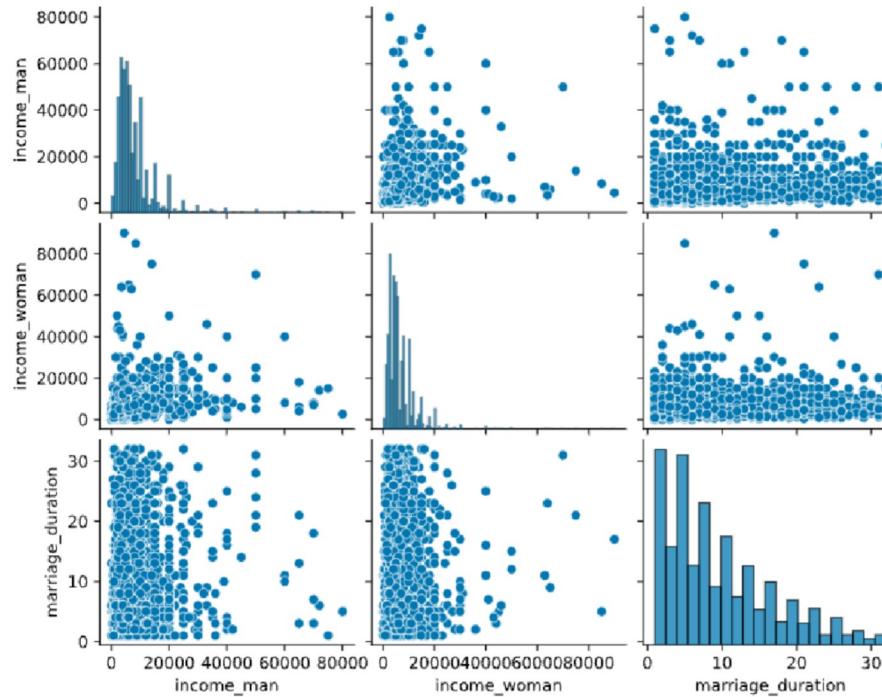
Pairplots

```
sns.pairplot(data=divorce)  
plt.show()
```



Pairplots

```
sns.pairplot(data=divorce, vars=["income_man", "income_woman", "marriage_duration"])
plt.show()
```



Let's practice!

EXPLORATORY DATA ANALYSIS IN PYTHON

